AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions and listings of claims in the application:

1. (Currently amended) A method for accessing information in a memory, comprising:

providing virtual address information to a memory management unit;

obtaining, from the memory management unit, a key tag and physical address information corresponding to the virtual address information;

retrieving a secret key using the key tag when it is determined based on the key tag that a memory location corresponding to the physical address information is protected; and

decrypting information read from the memory location using the secret key.

2. (Original) The method of claim 1, the retrieving comprising:

looking up the secret key in a secret key table using the key tag based on a determination that the memory location is protected.

3. (Original) The method of claim 1, further comprising:

writing unencrypted data to the memory location based on a determination that the memory location is unprotected.

4. (Original) The method of claim 1, further comprising:

reading unencrypted data from the memory location based on a determination that the memory location is unprotected.

5. (Original) The method of claim 1, further comprising:

executing an unencrypted instruction from the memory location based on a determination that the memory location is unprotected.

6. (Original) The method of claim 1, wherein the decrypted information is an instruction, further comprising:

executing the instruction.

7. (Original) The method of claim 1, wherein the decrypted information is data.

8. (Currently amended) The method of claim 1, further comprising:

encrypting data written to the ~~first~~ memory location using the secret key.

9. (Currently amended) A method for accessing information in a memory, comprising:

providing a virtual address to a memory management unit;

obtaining a key tag and a physical address corresponding to the virtual address from the memory management unit;

determining, based on the key tag, whether a memory location corresponding to the physical address is protected;

accessing a secret key in a secret key table using the key tag if it is determined that the memory location is protected; and

decrypting information read from [[a]] the memory location ~~corresponding to the physical address~~ using the secret key.

10. (Currently amended) A method for accessing information in a memory, comprising:

receiving a virtual address from a processor;

retrieving a key tag and a physical address corresponding to the virtual address; ~~and~~

determining based on the key tag whether a memory location corresponding to the physical address is protected; and

providing the key tag and the physical address to the processor, wherein a secret key associated with the key tag is used to decrypt information read from [[a]] the

memory location ~~corresponding to the physical address~~ if it is determined that the memory location is protected.

11. (Currently amended) The method of claim 10, wherein the secret key is used to encrypt data written to the ~~first~~ memory location.

12. (Original) The method of claim 10, wherein the decrypted information is data.

13. (Original) The method of claim 10, wherein the decrypted information is an instruction.

14. (Original) The method of claim 10, the retrieving comprising:

looking up the key tag in a memory mapping table using the virtual address information.

15. (Original) A method for accessing information in a memory, comprising:

receiving, at a memory management unit, virtual address information from a processor;

retrieving a key tag and physical address information corresponding to the virtual address information;

sending, from the memory management unit to the processor, a key tag and physical address information corresponding to the virtual address information;

determining whether a memory location corresponding to the physical address information is protected based on the key tag;

accessing a secret key in a secret key table using the key tag based on the determining; and

decrypting information read from the memory location using the secret key.

16. (Original) A method for loading encrypted information into a memory, comprising:

determining whether a header associated with a program block includes an encrypted secret key;

decrypting the encrypted secret key to form a decrypted secret key when a result of the determination indicates that the header includes an encrypted secret key;

storing the decrypted secret key in a secret key table;

assigning the decrypted secret key a key tag for use in retrieving the decrypted secret key from the secret key table;

loading the program block into the memory at a first memory location; and

associating the key tag with virtual address information and physical address information corresponding to the memory location, wherein information read from the first memory location is decrypted using the decrypted secret key.

17. (Original) The method of claim 16, further comprising:

validating a signature on the decrypted secret key before storing the decrypted secret key in the secret key table.

18. (Original) The method of claim 16, further comprising:

providing a key tag indicating that the program block is unencrypted based on a determination that the header does not include the encrypted secret key;

loading the unencrypted program block into the memory at a second memory location; and

associating the key tag indicating that the program block is unencrypted with virtual address information and physical address information corresponding to the second memory location.

19. (Original) The method of claim 16, wherein the decrypted information is an instruction.

20. (Original) The method of claim 16, wherein the decrypted information is data.

21. (Original) The method of claim 16, wherein data written to the first memory location is encrypted using the decrypted secret key.

22. (Currently amended) An apparatus for accessing information in a memory, comprising:

means for providing virtual address information to a memory management unit;

means for obtaining, from the memory management unit, a key tag and physical address information corresponding to the virtual address information;

means for retrieving a secret key using the key tag when it is determined based on the key tag that a memory location corresponding to the physical address information is protected; and

means for decrypting information read from the memory location using the secret key.

23. (Original) The apparatus of claim 22, the means for retrieving comprising:

means for looking up the secret key in a secret key table using the key tag based on a determination that the memory location is protected.

24. (Original) The apparatus of claim 22, further comprising:

means for writing unencrypted data to the memory location based on a determination that the memory location is unprotected.

25. (Original) The apparatus of claim 22, further comprising:

means for reading unencrypted data from the memory location based on a determination that the memory location is unprotected.

26. (Original) The apparatus of claim 22, further comprising:

means for executing an unencrypted instruction from the memory location based on a determination that the memory location is unprotected.

27. (Original) The apparatus of claim 22, wherein the decrypted information is an instruction, further comprising:

means for executing the instruction.

28. (Original) The apparatus of claim 22, wherein the decrypted information is data.

29. (Currently amended) The apparatus of claim 22, further comprising:

means for encrypting data written to the ~~first~~ memory location using the secret key.

30. (Currently amended) An apparatus for accessing information in a memory, comprising:

means for receiving virtual address information from a processor;

means for retrieving a key tag and physical address information corresponding to the virtual address information; ~~and~~

means for determining based on the key tag whether a memory location corresponding to the physical address information is protected; and

means for providing the key tag and physical address information to the processor, wherein a secret key associated with the key tag is used to decrypt information read from [[a]] the memory location ~~corresponding to the physical address information~~ when it is determined that the memory location is protected.

31. (Currently amended) The apparatus of claim 30, wherein the secret key is used to encrypt data written to the ~~first~~ memory location.

32. (Original) The apparatus of claim 30, wherein the decrypted information is data.

33. (Original) The apparatus of claim 30, wherein the decrypted information is an instruction.

34. (Original) The apparatus of claim 30, the means for retrieving comprising:

means for looking up the key tag in a memory mapping table using the virtual address information.

35. (Original) An apparatus for loading encrypted information into a memory, comprising:

means for determining whether a header associated with a program block includes an encrypted secret key;

means for decrypting a secret key based on a result of the determination;

means for storing the decrypted secret key in a secret key table;

means for assigning the decrypted secret key a key tag for use in retrieving the decrypted secret key from the secret key table;

means for loading the program block into the memory at a first memory location; and

means for associating the key tag with virtual address information and physical address information corresponding to the memory location, wherein information read from the first memory location is decrypted using the decrypted secret key.

36. (Original) The apparatus of claim 35, further comprising:

means for validating a signature on the decrypted secret key before storing the decrypted secret key in the secret key table.

37. (Original) The apparatus of claim 35, further comprising:

means for providing a key tag indicating that the program block is unencrypted based on a determination that the header does not include the encrypted secret key;

means for loading the unencrypted program block into the memory at a second memory location; and

means for associating the key tag indicating that the program block is unencrypted with virtual address information and physical address information corresponding to the second memory location.

38. (Original) The apparatus of claim 35, wherein the decrypted information is an instruction.

39. (Original) The apparatus of claim 35, wherein the decrypted information is data.

40. (Original) The apparatus of claim 35, wherein data written to the first memory location is encrypted using the decrypted secret key.

41. (Original) A computer-readable medium containing instructions for performing a method for accessing information in a memory, the method comprising:

receiving, at a memory management unit, virtual address information from a processor;

retrieving a key tag and physical address information corresponding to the virtual address information;

sending, from the memory management unit to the processor, a key tag and physical address information corresponding to the virtual address information;

determining whether a memory location corresponding to the physical address information is protected based on the key tag;

accessing a secret key in a secret key table using the key tag based on the determining; and

decrypting information read from the memory location using the secret key.

42. (Original) A computer-readable medium containing instructions for performing a method for loading encrypted information into a memory, the method comprising:

determining whether a header associated with a program block includes an encrypted secret key;

decrypting the encrypted secret key to form a decrypted secret key when a result of the determination indicates that the header includes an encrypted secret key;

storing the decrypted secret key in a secret key table;

assigning the decrypted secret key a key tag for use in retrieving the decrypted secret key from the secret key table;

loading the program block into the memory at a first memory location; and

associating the key tag with virtual address information and physical address information corresponding to the memory location, wherein information read from the first memory location is decrypted using the decrypted secret key.

43. (Original) An apparatus for accessing information in a memory, comprising:

a processor; and

a memory management unit operable to receive a virtual address from the processor, retrieve a key tag and a physical address corresponding to the virtual address, and send the key tag and physical address to the processor,

wherein the processor receives the key tag and physical address corresponding to the virtual address, determines whether a memory location corresponding to the physical address is protected based on the key tag, retrieves a secret key using the key tag based on the determining, and decrypts information read from the memory location using the secret key.

44. (Original) The apparatus of claim 43, wherein the processor writes unencrypted data to the memory location based on a determination that the first memory location is unprotected.

45. (Original) The apparatus of claim 43, wherein the processor reads unencrypted data from the memory location based on a determination that the first memory location is unprotected.

46. (Original) The apparatus of claim 43, wherein the processor executes an unencrypted instruction from the memory location based on a determination that the first memory location is unprotected.

47. (Original) The apparatus of claim 43, wherein the decrypted information is an instruction and the processor executes the instruction.

48. (Original) The apparatus of claim 43, wherein the decrypted information is data.

49. (Original) The apparatus of claim 43, wherein the processor encrypts data written to the memory location using the secret key.

50. (Original) An apparatus for loading encrypted information into a memory, comprising:

a memory including a program that: determines whether a header associated with a program block includes an encrypted secret key; decrypts the encrypted secret key to form a decrypted secret key when a result of the determination indicates that the header includes an encrypted secret key; stores the decrypted secret key in a secret key table; assigns the decrypted secret key a key tag for use in retrieving the decrypted secret key from the secret key table; loads the program block into the memory at a memory location; and associates the key tag with virtual address information and physical address information corresponding to the memory location, wherein information read from the memory location is decrypted using the decrypted secret key; and

a processor that runs the program.

51. (Currently amended) A method for protecting information in a memory, comprising:

generating a secret key in response to instructions from a program;

storing the secret key in a secret key table;

assigning the secret key a key tag for use in retrieving the secret key from the secret key table; and

associating the key tag with virtual address information and physical address information corresponding to a memory location of a program block from the program;

determining whether a memory location corresponding to the physical address information is encrypted based on the key tag, wherein information read from the memory location is decrypted using the secret key if it is determined that the memory location is encrypted.